

Outils de Programmation 2

Chapitre 3 : Vecteurs et matrices

Scilab était conçu à l'origine pour permettre aux mathématiciens, scientifiques et ingénieurs d'utiliser facilement les mécanismes de l'algèbre linéaire. Par conséquent, l'utilisation des vecteurs et des matrices est très intuitive en Scilab.

1 Les vecteurs

- Un vecteur est une liste ordonnée d'objets (réel, complexe, chaîne de caractères, booléens). Si les éléments sont arrangés horizontalement on dit que le vecteur est un vecteur ligne, par contre si les éléments sont arrangés verticalement on dit que c'est un vecteur colonne.
- Pour créer un vecteur ligne il suffit de donner la liste de ses éléments entre crochets `[]`, les éléments sont séparés par des espaces ou des virgules.

Exemple.

```
-->v=[1 2 3 4]
v =
1.  2.  3.  4.
-->u=[-1,-2,-3]
u =
-1.  -2.  -3.
```

- Pour créer un vecteur colonne, on donne la liste de ses éléments entre `[]` séparés par des points virgules, ou bien, on calcule sa transposé.

Exemple.

```
-->A=[5;-2;3]
A=
5
-2
3
-->B=[0 -1 2]'
B=
0
-1
2
```

1.1 Création d'un vecteur de suite arithmétique

Il existe deux méthodes permettant de créer un vecteur où ses éléments forme une suite arithmétique.

Méthode 1 :

Si les éléments d'un vecteur X sont ordonnés avec des valeurs consécutives, nous pouvons les noter par :
 $X=V0:P:Vf$

cette commande crée un vecteur ligne dont le premier élément est V0 et le dernier est Vf et le pas P (raison de la suite).

Exemple.

```
-->X=0:2:10
```

```
X =
```

```
0.  2.  4.  6.  8.  10.
```

```
-->X=0:0.5:2
```

```
X =
```

```
0.  0.5  1.  1.5  2.
```

Méthode 2 : (commande linspace)

Cette méthode permet de créer un vecteur de suite arithmétique en précisant la valeur initiale V0, la valeur finale Vf et le nombre de terme n. Scilab calcule automatiquement le pas $P = \frac{Vf-V0}{n-1}$.

```
X= linspace(V0,Vf,n)
```

Exemple.

```
-->X=linspace(0,100,5)
```

```
X =
```

```
0.  25.  50.  75.  100.
```

1.2 Manipulation et accès aux éléments d'un vecteur

L'accès aux éléments d'un vecteur se fait en utilisant la syntaxe suivante :

```
nom_vecteur(position)
```

position: peut être un simple indice, ou un vecteur d'indice.

Exemples.

```
-->X=[1,-1,2,0.5,3*%i]
```

```
X =
```

```
1.  -1.  2.  0.5  3.i
```

```
-->X(4) //la 4ième position
```

```
ans =
```

```
0.5
```

```
-->X(3:5) //de la troisième position jusqu'au cinquième
```

```
ans =
```

```
2.  0.5  3.i
```

```
ans =
```

```
-->X(5:-2:1) //de la cinquième position jusqu'au première avec un pas = -2
```

```
ans =
```

```
3.i  2.  1.
```

```
-->X(2:$) //de la deuxième position jusqu'à la dernière
```

```
ans =
```

```
-1.  2.  0.5  3.i
```

```
-->X([1,3,5]) //la 1ière, la 3ième et la 5ième position
```

```
ans =
```

```
1.  2.  3.i
```

```

-->X(2)=5 //remplacer la deuxième valeur par 5
ans =
1. 5. 2. 0.5 3.i
-->X(6)=-4 //ajouter un 6ième élément de valeur -4
ans =
1. 5. 2. 0.5 3.i -4
-->X(10)=7 //ajouter un 10ième élément avec la valeur 7
ans =
1. 5. 2. 0.5 3.i -4. 0. 0. 0. 7.
-->X(2)=[] //supprimer le deuxième élément
ans =
1. 2. 0.5 3.i -4. 0. 0. 0. 7.
--> X(6:8)=[] //supprimer du 6ième jusqu'au 8ième élément
ans =
1. 2. 0.5 3.i -4. 7.
-->X(:)=2 //affecter tous les éléments par une seule valeur
ans =
2. 2. 2. 2. 2. 2.
-->X($)=X($)*4 //multiplier le dernier élément par 4
ans =
2. 2. 2. 2. 2. 4.
-->X(X<4) //donner tout les valeurs inférieur à 4
ans =
2. 2. 2. 2. 2.

```

1.3 Opérations sur les vecteurs

On considère les deux vecteurs :

```
-->X=[1 -2 3]
```

```
-->Y=[-4 5 6]
```

Opération	Exemple	
'+' l'addition	-->X+2 ans = 3. 0. 5.	-->X+Y ans = -3. 3. 9.
'-' soustraction	-->X-1 ans = 0. -3. 2.	-->X-Y ans = 5. -7. -3.
'.*' multiplication élément par élément	-->X*2 ans = 2. -4. 6.	-->X.*Y ans = -4. -10. 18.
'./' division élément par élément	-->X/2 ans = 0.5 -1. 1.5	-->X./Y ans = -0.25 -0.4 0.5
'.^' puissance élément par élément	-->X.^2 ans = 1. 4. 9.	-->X.^Y ans = 1. -32. 729.

Remarque. X^2 génère une erreur car cette expression réfère a une multiplication de deux matrices. En effet (X*X, doit être réécrite X*X' ou X'*X pour être valide)

1.4 Fonctions spécifiques aux vecteurs

Scilab implémente un ensemble de fonctions spécifiques au traitement des vecteurs.

Fonction	Signification
<code>sum(X)</code>	somme des éléments du vecteur X
<code>prod(X)</code>	produit des éléments du vecteur X
<code>max(X)</code>	plus grand élément du vecteur X
<code>min(X)</code>	plus petit élément du vecteur X
<code>mean(X)</code>	moyenne des éléments du vecteur X
<code>gsort(X,'g','i')</code> , <code>gsort(X,'g','d')</code>	retourne un vecteur trié en ordre croissant (décroissant)

1.5 Vecteurs spéciaux

Vecteur	Signification
<code>ones(1,n)</code> , <code>ones(m,1)</code>	vecteur ligne (colonne) de longueur n, (m) dont les éléments valent 1.
<code>zeros(1,n)</code> , <code>zeros(m,1)</code>	vecteur ligne (colonne) de longueur n, (m) dont les éléments valent 0
<code>rand(1,n)</code> , <code>rand(m,1)</code>	les éléments du vecteur sont générés de manière aléatoire entre 0 et 1